SQL Fundamenenenentals

# Các kiểu dữ liệu trong SQL – ORACLE

Các kiểu ký

* Số ( int, float ...)
* Ký tự ( varchar, varchar2 ...)
* Ngày tháng (date)
  + Mặc định: DD-MON-YY

Chuyển đổi dữ liệu giữa các loại dữ liệu:

* Sang kiểu ký tự :

TO\_CHAR(hire\_date, 'J')

* Sang kiểu ngày tháng

TO\_DATE ('November 13, 1992', 'MONTH DD, YYYY')

Phân biệt NULL , 0 và kí tự trắng ( blank space)

* NULL là kiểu kí tự chưa xác định
* 0 là kí tự dạng số
* Kí tự trắng là kí tự

# Mệnh đề SELECT

**SELECT** là mệnh đề sử dụng để lựa chọn dữ liệu ( column, row ) từ các đối tượng ( table, view..) với các điều kiện xác định ( where, having ...)

Cú pháp:

SELECT \*|{[DISTINCT] *column|expression* [*alias*],...}

FROM *table*

[WHERE *condition(s)*];

Các nguyên tắc khi viết mệnh đề **SELECT**:

* Viết hoa từ khóa ( SELECT, FROM, WHERE...)
* Các từ khác viết thường
* Có thể viết trên nhiều dòng, mỗi dòng bắt đầu bằng 1 từ khóa
* Kết thúc mệnh đề bằng dấu **“** **; ”**

Trước khi sử dụng SELECT thì nên xem cấu trúc của đối tượng trước

DESC departments;

Các tùy chọn khi **SELECT:**

* SELECT tất cả dữ liệu của đối tượng :

SELECT \*

FROM departments;

* Chỉ rõ các column được SELECT:

SELECT department\_id, location\_id

FROM departments;

* Có thể thêm biểu thức số học ở SELECT

SELECT last\_name, salary, salary + 300

FROM employees;

* + Chỉ các trường dữ liệu kiểu số hoặc ngày tháng thì mới có biểu thức số học
  + Các biểu thức số học bao gồm: **“** **+ - \* / ”** và các dấu ngoặc
* Sử dụng Distinct để loại bỏ các kết quả trùng lặp.
  + Từ khóa Distinct có ý nghĩa đối với cả mệnh đề SELECT

SELECT DISTINCT department\_id, lastname

FROM employees;

* Sử dụng Alias để thay đổi nội dung hiện thị

SELECT last\_name AS name, commission\_pct comm

FROM employees;

* + Trong trường hợp có kí tự đặc biệt thì phải cho tên hiện thì vào dấu nháy

SELECT last\_name "Name" , salary\*12 "Annual Salary"

FROM employees;

* Sử dụng kí tự nối
  + Tạo dấu “ \_ ” giữa các trường

SELECT last\_name||job\_id AS "Employees"

FROM employees;

* + Thêm vào giữa các trường các kí tự khác

SELECT last\_name ||' is a '||job\_id

AS "Employee Details"

FROM employees;

* Sử dụng ORDER BY để sắp xếp kết quả được hiện thì
  + ASC: Ascending order, default
  + DESC: Descending order

SELECT last\_name, job\_id, department\_id, hire\_date

FROM employees

ORDER BY hire\_date ;

# Giới hạn kết quả trong biểu thức SQL

## Biểu thức điều kiện trên từng row

Mệnh đề WHERE được sử dụng để giới hạn kết quả khi thực hiện. Mệnh đề này làm việc theo cơ chế kiểm tra dữ liệu trên từng dòng ( row)

Cú pháp:

SELECT \*|{[DISTINCT] *column|expression* [*alias*],...}

FROM *table*

WHERE *condition(s)*;

### Các biểu thức điều kiện trong WHERE:

Các biểu thức điều kiện trong WHERE bao gồm:

**= , <, <=, >, =>, <>, BETWEEN ...AND..., IN , LIKE , IS NULL , IS NOT NULL**

Biểu thức điều kiện bằng

SELECT employee\_id, last\_name, job\_id, department\_id

FROM employees

WHERE department\_id = 90 ;

Đối với ký tự thì cần đưa vào dấu ngoặc đơn ‘’

SELECT last\_name, job\_id, department\_id

FROM employees

WHERE last\_name = 'Whalen' ;

Trong mệnh đề WHERE, có thể sử dụng các biểu thức logic : AND, OR, NOT và các phép toán như MINUS ( trừ ), INTERSECT( giao), UNION ( hợp )

SELECT last\_name, job\_id, salary

FROM employees

WHERE job\_id = 'SA\_REP'

OR job\_id = 'AD\_PRES'

AND salary > 15000;

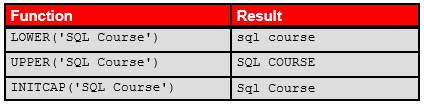
...

### Sử dụng Single-Row Functions

**Các hàm đối với dữ liệu kiểu ký tự ( character)**

**Case-Conversion Functions** là các hàm convert các chuỗi thành các kí tự tương đương:

* LOWER: tất cả các kí tự thành chữ cái thường
* UPPER: tất cả các kí tự thành chữ cái hoa
* INITCAP: chữ cái đầu tiên của mỗi từ được viết hoa

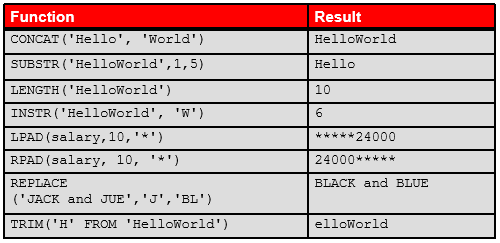


Sử dụng **Case-Conversion Functions:**

SELECT employee\_id, last\_name, department\_id

FROM employees

WHERE LOWER(last\_name) = 'higgins';

**Character-Manipulation Functions** :

Ví dụ:

SELECT employee\_id, CONCAT(first\_name, last\_name) NAME,

job\_id, LENGTH (last\_name),

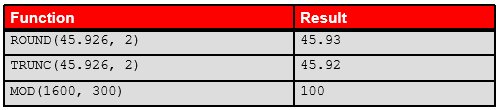
INSTR(last\_name, 'a') "Contains 'a'?"

FROM employees

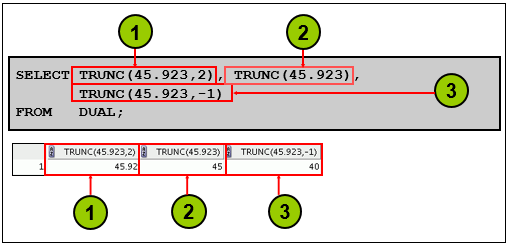
WHERE SUBSTR(job\_id, 4) = 'REP';

**Các hàm đối với dữ liệu kiểu số:**

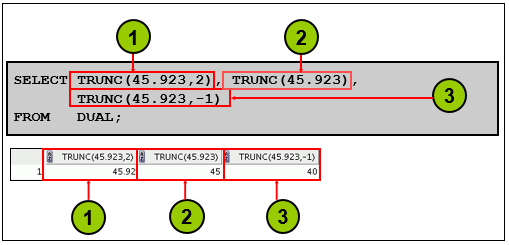
* ROUND: Làm tròn dữ liệu lên
* TRUNC: Làm tròn dữ liệu xuống
* MOD: Chia lây phần dư



Sử dụng ROUND:



Sử dụng TRUNC:



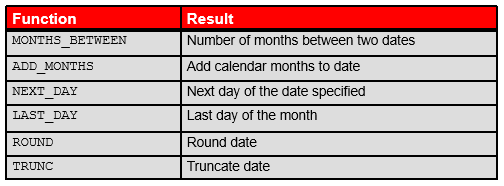
Sử dụng MOD

SELECT last\_name, salary, MOD(salary, 5000)

FROM employees

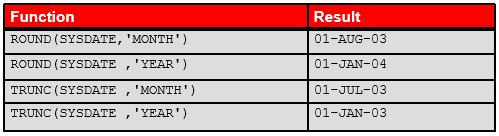
WHERE job\_id = 'SA\_REP';

**Các hàm đối với dữ liệu kiểu date:**



Ví dụ:

* SYSDATE = '25-JUL-03'



## Biểu thức điều kiện trên nhiều row

* Select có cái gì thì Group by có cái đó
* Nếu có nhiều group by thì
  + Các dòng được group by theo mệnh đề thứ nhất
  + Sau đó được group by theo mệnh đề thứ 2

# Using Conversion Functions and Conditional Expressions

# Reporting Aggregated Data Using the Group Functions

# Displaying Data from Multiple Tables Using Joins